Amendments to the Specification

Please replace paragraph **[0007]** on page 5 with the following amended paragraph:

[0007] In accordance with yet another aspect of the invention, a method is provided for communicating private data between computers coupled to a data communication network. The method includes receiving, at a server, a request from a roaming client for encrypted private data. The request includes a digest or hashed value of an authentication password. The server and the roaming client are coupled to the data communication network. The method further includes determining if a form of the authentication password received from the roaming client is valid. The method further includes retrieving, when a form of the authentication password is valid, the encrypted private data, where the private data was previously encrypted as a function of ~~an encryption password~~ a wrapping key unknown to the server. The method further includes transferring the retrieved encrypted private data from the server to the roaming client for decryption as a function of the wrapping key.

Please replace paragraph **[0024]** on page 5 with the following amended paragraph:

[0024] Referring now to FIG. 1, a block diagram illustrates an exemplary network environment 100 having a home or local client computer 102 ~~is~~ coupled to a data communication network 104. In this example, the network 104 is the Internet (or the World Wide Web). However, the teachings of the present invention can be applied to any data communication network. Multiple roaming or remote client computers 106 and 108 are also coupled to network 104. In turn, the home client computer 102 can communicate with roaming client computers 106 and 108 via network 104. Home client computer 102 denotes a computer that the user has a reason to trust because un-trusted third parties do not have unrestricted or easy physical access to it. Roaming client computers 106 and 108 denote computers in which the user has more limited trust because it is owned and physically controlled by a third party. An authentication server 110 coupled to network 104 allows communication between itself and the home client computer 102 and roaming client computers 106 and 108. Although referred to as an "authentication server,"

authentication server 110 in the illustrated embodiment is simply a web server capable of authenticating users as well as capable of interacting with web browsers and other web servers.

Please replace paragraph **[0030]** on page 6-7 with the following amended paragraph:

[0030] A home client application 206 allows a home user 208 to encrypt private data and transfer the encrypted private data to the web server 204 via a communication network 212 (e.g., network 104). The home client application 206 is executable by the home client 202 and responsive to user input for initiating the encryption of the private data, and the transfer of the encrypted private data to the web server 204. In this embodiment, the home client application 206 includes a memory 214 storing a private key 216 used during a public key encryption process such as described above in reference to FIG. 1. The home client application 206 includes an encryption algorithm 218 for performing a mathematical operation on the private data to convert it into encrypted private data. More specifically, the encryption algorithm 218 is used in conjunction with key data to convert the private data. As known to those skilled in the art, a number of encryption algorithms (e.g., 3DES algorithm and HMAC-RC4™ algorithm) can be used to encrypt data such that it is nearly impossible to decrypt the content without knowledge of the encryption key.

Please replace paragraph **[0032]** on page 7 with the following amended paragraph:

[0032] A user-interface (UI) 220 linked to the home client 202 allows the user 208 to interact with the web server 204. For example, the UI ~~214~~ 220 may include a display 217 such as a computer monitor for viewing data and/or input forms, and an input device 219 such as a keyboard or a pointing device (e.g., a mouse, trackball, pen, or touch pad) for entering data into the input form ~~(not shown)~~ 210. In other words, UI 214 allows user 208 to select data on the home client ~~210~~ 202 for encryption, and allows user 208 to submit a request to transfer encrypted data from the home client 202 to the web server 204 for storage.

Please replace paragraph **[0033]** on page 7 with the following amended paragraph:

[0033] A database 226 (e.g., database ~~110~~ 112) is coupled to web server 204 and contains information necessary to validate a request from the home client 202 (as well as other users on the network) to store encrypted private data in database 226. Although database ~~210~~ 226 is shown separately from authentication server 204, it is to be understood that in other embodiments of the invention, database ~~210~~ 226 may be contained within web server ~~208~~ 204.

Please replace paragraph **[0039]** on page 9 with the following amended paragraph:

[0039] In this embodiment, the server roaming application 228 receives a request from the roaming client 302 via roaming client application 304 and communication network 212 to retrieve stored encrypted private data from the database 226. The server roaming application 228 is responsive to the received request, and executable by web server 204, for authenticating roaming user 306. In this embodiment, server 204 requests the authentication password from roaming user 306 via an input form 210 such as shown in FIG. 2[[A]].

Please replace paragraph **[0040]** on page 9 with the following amended paragraph:

[0040] In substantially the same manner as described above in reference to FIG. 2, the server roaming application 228 validates a form of the authentication password received from the client to determine if the roaming user 306 is authorized to retrieve encrypted data from the database. If the authentication password is not validated, the server roaming application 228 denies the roaming client 302 access to the encrypted private data stored in the database 226. Alternatively, if the authentication password is validated, the server roaming application 228 retrieves encrypted data from the database 226, as indicated by reference character 310, and transfers the encrypted data to the roaming client 302, as indicated by reference character 311. The roaming client application 304 is responsive to the received encrypted private data to request the encryption password ~~308~~ from the user 306 and generate the wrapping key K1 and to execute a decryption algorithm 312. ~~(308 is depicted incorrectly in figure 3.)~~ In this case, the decryption

algorithm 312 decrypts the received encrypted private data as a function of the wrapping key 230 generated on the roaming client 302 to obtain the private key associated with the home client 202. Thereafter, the roaming client application 304 can store the obtained private key in a memory ~~314~~ 214 associated with the roaming client 302.

Please replace paragraph **[0042]** on page 9 with the following amended paragraph:

[0042] In this embodiment, the home user 208 uses the UI 220 to select a recovery option to have the ability to recover encrypted private data stored on the server 204 even when the user cannot remember the encryption password. In this embodiment, in addition to generating the wrapping key K1, as a function of the encryption password, as described above in reference to FIG. 2, ~~roaming~~ home client application 212 is responsive to the recovery key request, to randomly generate a new encryption key (i.e., recovery key K2). For instance, after user 208 has input the encryption password into a form (not shown), the user 208 uses, for example, a mouse to click a "YES" button presented to the user along with a dialog box displaying the message "ENABLE PRIVATE DATA RECOVERY WITHOUT ENCRYPTION PASSWORD." The home client application 206 is responsive to to a "YES" selection by the user to randomly generate the recovery key K2. Notably, K2 is not linked in any way to the encryption password. The server roaming application 228 authenticates user 208 by validating the authentication password received from the client. If the authentication password is validated, the user 208 is allowed to transfer encrypted private data including a private key encrypted with the wrapping key $E_{K1}PK$ 234, a wrapping key encrypted with the recovery key $E_{K2}K1$ 414, and a recovery key encrypted with the wrapping key $E_{K1}K2$ 416 to the server 204, as indicated by 418, 419, and 420 respectively. The server roaming application 228 is responsive to the received encrypted private data to store $E_{K1}PK$, $E_{K2}K1$ 414, and $E_{K1}K2$ in the database 226, as indicated by 421, 422, and 423 respectively. Moreover, the server roaming application 228 is responsive to encrypted data received from the home client 202 to randomly generate a backup key K3 424 for storage in the database 226, as indicated by 425, and to transfer the generated backup key 424 to the home client 202, as indicated by 426. Note that the transfer of K3 occurs via a secure channel (e.g., over SSL.) Without a secure channel, the value could be changed so that the

recovery key can be easily discovered. The home client application 206 is responsive to the received backup key 424 to generate a second encrypted recovery key $E_{K3}K2$ 428 to store in memory 214 and/or on a disk associated with the home client 202.

Please replace paragraph **[0046]** on pages 11-12 with the following amended paragraph:

[0046] The roaming client application 304 is responsive to the received first encrypted recovery key 416 to request the encryption password 504 from the user, generate the wrapping key K1, and execute the decryption algorithm 312. In this case, the decryption algorithm 312 decrypts the received first encrypted recovery key 416 as a function of the wrapping key 230 generated on the roaming client 302 to obtain the recovery key 408 associated with the home client computer 202. The roaming client application 304 is responsive to the received backup key 424 to execute the encryption algorithm 218. In this case, the encryption algorithm 218 encrypts the obtained recovery key 408 as a function of the received backup key 424 to generate the second encrypted recovery key 428. Thereafter, the roaming client application 304 stores the second encrypted recovery key 428 in the memory ~~314~~ 214 and onto disk associated with the roaming client 302. As a result of having the second encrypted recovery key 428 stored in memory ~~314~~ 214, the backup key 424 and the encrypted wrapping key from the server, the roaming client 302 can recover and decrypt an encrypted private key 234 being stored in database 226 without having knowledge of a password (i.e., encryption password) used to generate the wrapping key K1.

Please replace paragraph **[0048]** on page 12 with the following amended paragraph:

[0048] In this embodiment the backup process described above has been preformed on the roaming client 302 so that roaming client 302 has second encrypted recovery key 428 in memory ~~314~~ 214. The server roaming application 228 receives a request from a roaming client 302 via roaming client application 304 and communication network 212 to retrieve encrypted private data such as a private key 216 from the database 226. The server roaming application 228 authenticates user 302 by validating the authentication password received from the client.

When the user is successfully authenticated, the server application transfers the encrypted private data 216 to the roaming client application 304. As described above in reference to FIGS. 3 and 5, the roaming client application 304 is responsive to the encrypted private data to request the encryption password from the roaming user 306 in order to generate the wrapping key K1.

Please replace paragraph **[0055]** on page 14 with the following amended paragraph:

[0055] At 902 the user of the home client computer executes a roaming client application and designates private data stored in a memory of the home client computer to encrypt, and submits a request to transfer the designated private data to the server. If the request is authenticated as discussed above in reference to FIG. 8, the server executes a roaming server application at 904. The server roaming application determines if this is the first time the SRA is executed in response to a request from the particular home client at 906. If the server roaming application determines this is first execution at 906 or that a backup key K3 cannot be found in the database for this user, the server roaming application generates a random backup key K3 at 908. At 910, the server roaming application stores the backup key K3 in a database, and provides the backup key K3 to the home client application for use in encrypting recovery data at 912. If the server roaming application determines this is not the first execution at ~~907~~ 906, the server roaming application retrieves the backup key from the database at 909, and provides the backup key K3 to the home client application for use in encrypting recovery data at 912.

Please replace the section of the patent application referred to as "ABSTRACT OF THE INVENTION" with the following section:

ABSTRACT ~~OF THE INVENTION~~

~~A system and method for securely~~ Securely roaming private data from ~~a first~~ one client computer to another ~~second client computer linked via a communication~~ in a network. A ~~user of the first client computer executes a home client application and designates private data for roaming. The~~ home client application generates a first key in response to a password, and encrypts ~~the~~ designated private data as a function of the first key. ~~The~~ A server receives and

stores the encrypted private data. A ~~user of the second computer executes a roaming client application and requests transfer of the encrypted private data from the server. The~~ roaming client application generates the first key in response to the password, and decrypts encrypted private data transferred from the server to obtain the private data. The invention further provides users the ability to retrieve encrypted private from the server even when the user cannot remember the password associated with the first key. Also, the server has no knowledge of the private data ~~nor~~ or the keys.